

REMARKS/ARGUMENTS

This Response to Office Action is responsive to the Final Office Action dated September 8, 2006. This Response is filed within six months of the mailing date of the Office Action.

Applicants respectfully request further examination and reconsideration in view of the arguments set forth fully below. Claims 1-7, 13-25 are pending in this application. Claims 23-25 are new.

112 Rejections

The Examiner rejected claim 5 as failing to comply with the written description requirement. The applicants respectfully disagree, but in the interests of expediting issuance of a patent in this case have amended claim 5. The applicants reserve the right to reintroduce the original claim at a later date, if desired.

103 Rejections

Within the Office Action, Claims 1-7, 13-22 have been rejected under 35 U.S.C. § 103(a) as being allegedly unpatentable over U.S. Patent No. 5,991,543 to Amberg et al. (Amberg), U.S. Patent No. 5,854,924 to Rickel et al. (Rickel), further in view of Papachristou et al. (Papachristou) "Microprocessor Based Testing for Core-Based System on a Chip," 1999, ACM 1-58113-092-9/99/0006. The applicants respectfully disagree with this rejection.

Prior Art

Amberg apparently discloses a method for installing and/or testing software for a built-to-order computer system having a plurality of components including a plurality of elements. (Abstract). Amberg discloses testing of the customer's actual build-to-order computer system built by the manufacturer before it is shipped to the customer. "Before the target system 160 is shipped to the customer, the plurality of components are installed and tested." (Col. 3, lines 37-39).

Notably, Amberg does not disclose a simulation computer. Moreover, it is not **the process** that downloads and installs customer ordered software onto a target computer

that is tested. Rather, the **target computer** itself is tested. As the Examiner himself has pointed out at page 4 of the Office Action, Amberg discloses: "A method for installing and/or testing software for a build-to-order computer system..." (Abstract).

As the Examiner admits at page 4 of the Office Action, Amberg does not disclose a simulation of the software download and installation process in a manufacturing environment. The Examiner has not asserted that Amberg discloses simulating a launch of download and installation of customer ordered software, recursive evaluation procedures, determining which of a combination of files from a library would be executed in a manufacturing environment, simulating a manufacturing environment, or analyzing an outcome of a simulation to determine possible syntax errors and possible flow errors. Instead, the Examiner has used Rickel and Papachristou to make up for these deficiencies.

Rickel apparently discloses a static debugging tool arranged to debug files without executing the program being debugged. (Abstract). The Examiner asserts that Rickel discloses analyzing the outcome of the simulation of the execution of a dynamically generated file to determine possible syntax errors and possible flow errors. However, contrary to the Examiner's assertion, Rickel does not disclose possible flow errors. Moreover, Rickel does not make a determination as to which files from a library would be executed in a computer manufacturing environment. Instead, Rickel discloses a decompiler that receives and disassembles a single binary file by creating an intermediate disassembled file.

Rickel does not disclose simulating a launch of download and installation of customer ordered software, recursive evaluation procedures, determining which of a combination of files from a library would be executed in a manufacturing environment, or simulating a manufacturing environment. Rickel does not disclose examining multiple files for recursive examination from libraries to determine which ones would be executed in the factory. It follows that Rickel does not disclose monitoring what these multiple files would do with the resources of an execution environment.

The Examiner relies upon Papachristou to make up for the fact that Amberg and Rickel do not disclose simulating the process of downloading a file. Importantly, Papachristou explains that "times can be calculated analytically or by simulating the download..." In other words, if one calculates the amount of time it takes to download, that is equivalent to simulating the download. Simulating the download to determine the length of the download phase is an entirely different issue from simulating **the process of downloading and installing** customer ordered software. Papachristou is concerned with the amount of time that it takes to download ("the raw download time"), not with the process of downloading itself.

Moreover, Papachristou is in an entirely different field from the prior art with which it is combined. Specifically, Papachristou discloses the development of a flexible design for test methodology for testing a core-based system on chip (SOC) where entire systems are designed onto a single semiconductor chip. Papachristou discloses a feature by using an embedded microprocessor/memory pair to test the remaining components on the SOC; Papachristou discloses improvements to the field of semiconductor technology. The improvements of Papachristou are measured in terms of saving **milliseconds**, as compared to improvements in a manufacturing environment which could be measured in terms of saving **hours** of repetitive testing on the factory floor. Thus, in addition to being in a different field of endeavor, Papachristou is not reasonably pertinent to the particular problem with which the claims are concerned.

The Examiner has not asserted that Papachristou discloses recursive evaluation procedures, determining which of a combination of files from a library would be executed in a manufacturing environment, simulating a manufacturing environment, or analyzing an outcome of a simulation to determine possible syntax errors and possible flow errors. Plus, for the reasons discussed above, Papachristou does not disclose simulating a launch of download and installation of customer ordered software.

Prior Art Distinguished

Claim 1 and Dependent Claims

Claim 1 includes the language "dynamically generating on a simulation computer a file that includes instructions that when executed launch a process of downloading and installing on a target computer customer ordered software, including a combination of files from a library that include source code responsible for downloading the customer ordered software onto the target computer[.]" As discussed above, Amberg, Rickel, and Papachristou, whether considered alone or in combination, do not teach simulating a process of downloading and installing on a target computer customer ordered software onto a target computer. (Note, claim 1 includes the language "simulating execution of said dynamically generated file.")

Claim 1 includes the language "simulating execution of said dynamically generated file through recursive evaluation procedures and in accordance with a set of evaluation rules to determine which of the combination of files from a library would be executed during a computer manufacturing process and interpret the outcome of the execution of said dynamically generated file[.]" As discussed above, Amberg, Rickel, and Papachristou, whether considered alone or in combination, do not disclose recursive evaluation procedures. Nor do Amberg, Rickel, and Papachristou disclose determining which of a combination of files from a library would be executed in a manufacturing environment.

Claim 1 includes the language "simulating a computer manufacturing environment within which the combination of files from the library run and interact[.]" As discussed above, Amberg, Rickel, and Papachristou, whether considered alone or in combination, do not disclose simulating a manufacturing environment.

Claim 1 includes the language "analyzing the outcome of the simulation of the execution of said dynamically generated file to determine possible syntax errors and possible flow errors." As discussed above, Amberg, Rickel, and Papachristou, whether considered alone or in combination, do not disclose analyzing an outcome of a simulation to determine possible syntax errors and possible flow errors. Specifically, the prior art of record do not disclose determining the possible flow errors.

To render a claim obvious, a reference must teach each and every element of the claim. The cited prior art fails to teach each and every element of claim 1 (and, in fact, fail to teach a great many of the elements of claim 1). Since the prior art references fail to teach each and every element of claim 1, claim 1 is allowable over the cited prior art.

Claims 2-4 and 13-20 are either directly or indirectly dependent on the independent claim 1. Accordingly, claims 2-4 and 13-20 are allowable as being dependent on an allowable claim, and potentially for other reasons as well.

Claim 5 and Dependent Claims

Claim 5 includes the language "a simulation computer, including a simulated computer manufacturing environment, that mimics a target computer[.]" As previously discussed, the cited prior art do not disclose a simulated computer manufacturing environment. Moreover, the Examiner has asserted, without support, that the simulation computer "mimics a target computer." The applicants respectfully request that the Examiner either provide support for this assertion or withdraw the rejection.

Claim 5 includes the language "a first process for creating a second process that launches a process that downloads and installs customer ordered software onto a target computer, including a combination of files from a library, wherein the library includes source code responsible for downloading the customer ordered software onto the target computer[.]" As previously discussed, the cited prior art do not disclose a simulating a process that downloads and installs customer ordered software onto a target computer. (Note, claim 5 includes the language "simulated execution of the second process.")

Claim 5 includes the language "a third process for examining said combination of files from the library through recursive evaluation procedures and in accordance with a set of evaluation rules to determine which files would be executed in the computer manufacturing environment and interpreting the outcome of simulated execution of the second process[.]" As previously discussed, the cited prior art do not disclose recursive evaluation procedures. Nor does the cited prior art disclose determining which of a combination of files from a library would be executed in a manufacturing environment.

Since the cited prior art fails to teach each and every element of claim 5, claim 5 is allowable over the cited prior art. Claims 21-22 are either directly or indirectly dependent on the independent claim 5. Accordingly, claims 21-22 are allowable as being dependent on an allowable claim and potentially for other reasons as well.

The New Claims

The applicants respectfully assert that the new claims 23-25 are allowable for reasons similar to those described with reference to claim 1.

Suggestion/Motivation to Combine

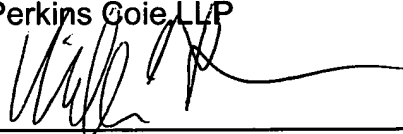
As was described above, Papachristou is believed to be nonanalogous art. In addition, the applicants respectfully assert that Amberg and Rickel do not include a suggestion or motivation to combine. Specifically, Amberg is directed to testing build-to-order computer systems, while Rickel is directed to a software programmer's debugging tool. The claims are not directed to debugging software, but rather to simulating a manufacturing environment, including download and installation of files on a computer system. Testing build-to-order systems (Amberg) and a software debugging tool (Rickel), combined, do not yield simulating a manufacturing environment without the use of hindsight reconstruction.

Conclusion

No new subject matter has been added by way of the above amendments. For the reasons given above, the applicants respectfully submit that Claims 1-7, 13-25 are in a condition for allowance. The applicant respectfully requests that all rejections be withdrawn and the application be allowed at the earliest date possible. Should the

Examiner have any questions or comments, he is encouraged to call the undersigned at (650) 838-4305 to discuss the same so that any outstanding issues can be expeditiously resolved.

Respectfully submitted,
Perkins Coie LLP



William F. Ahmann
Reg. No. 52,548

Date: December 8, 2006

Correspondence Address:

Customer No. 22918
Perkins Coie LLP
P.O. Box 2168
Menlo Park, California 94026
(650) 838-4300